Service Robots and Robotics:

Design and Application

Marco Ceccarelli
University of Cassino, Italy

ENGINEERING SCIENCE REFERENCE

Managing Director: Lindsay Johnston
Senior Editorial Director: Heather A. Probst
Book Production Manager: Sean Woznicki
Development Manager: Joel Gamon
Development Editor: Hannah Abelbeck
Acquisitions Editor: Erika Gallagher
Typesetter: Jen McHugh
Cover Design: Nick Newcomer, Lisandro Gonzalez

Published in the United States of America by
Engineering Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Service robots and robotics: design and application / Marco Ceccarelli, editor.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-4666-0291-5 (hardcover) -- ISBN 978-1-4666-0292-2 (ebook) -- ISBN 978-1-4666-0293-9 (print & perpetual access) 1. Robotics--Research. 2. Robotics--Design and construction. 3. Robots, Industrial--Research. 4. Robots, Industrial--Design and construction. I. Ceccarelli, Marco.
TJ211.S438 2012
629.8'92--dc23
2011044960

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 4

Mission Planning of Mobile Robots and Manipulators for Service Applications

Elias K. Xidias
University of the Aegean, Greece

Nikos A. Aspragathos
University of Patras, Greece

Philip N. Azariadis
University of the Aegean, Greece

ABSTRACT

The purpose of this chapter is to present a mission planning approach for a service robot, which is moving and manipulating objects in semi-structured and partly known indoor environments such as stores, hospitals, and libraries. The recent advances and trends in motion planning and scheduling of mobile robots carrying manipulators are presented. This chapter adds to the existing body of knowledge of motion planning for Service Robots (SRs), an approach that is based on the Bump-Surface concept. The Bump-Surface concept is used to represent the entire robot's environment through a single mathematical entity. Criteria and constraints for the mission planning are adapted to the service robots. Simulation examples are presented to show the effectiveness of the presented approach.

INTRODUCTION

In recent years, robot applications are moving from industrial environments to unstructured and/or semi-structured environments such as domestic and shop floors. Therefore, the service robot development is based on the rich heritage of the industrial robot research. However, the service robots should acquire new capabilities to perform in unstructured and/or semi-structured and partly-known environments with high-safety requirements since the robots are sharing the same workspace with people and other sensitive objects designed for human handling.

DOI: 10.4018/978-1-4666-0291-5.ch004

Service robots are called to perform fetch and carry tasks in domestic environments or professional environments varying from simple orders, e.g., "go to the refrigerator and fetch a bottle of milk," or more complicated tasks like serving food in a restaurant. In other professional environments, like market stores, libraries, and hospitals, transportation and courier tasks are usually required. These robots among other advanced capabilities should be able to plan their missions autonomously. This is the subject of this chapter.

The spectrum of service applications relevant to mission planning is growing continuously with the trend to take over the demand for industrial robot applications. In professional or domestic environments, the most promising robot applications that include mission-planning tasks are the following:

- Delivering medicine, food, and medical consumables in hospitals.
- Delivering or distributing books in libraries.
- Stores and pharmacy automation.
- Surveillance in dangerous areas.
- Materials distribution and delivery in construction sites.
- Helping people in domestic environments.
- Providing transport and delivery tasks in urban environments.

The missions of the service robots cannot be programmed a priori, as it usually happens in the applications with industrial robots. High level of autonomy, flexibility, and efficiency is required in partly known environments. In indoor environments, the objects are not located in a constant and predefined position and the demands can vary very often; therefore, the robots should be capable of planning and scheduling autonomously their collision free optimal routes. Since, most of the service applications include handling of objects; the motion planning of a manipulator, which is mounted, on a mobile platform represents one of the key issues in the research and development of service robots.

In this context, mission planning is identified as a fundamentally critical factor for an autonomous service robot among other capabilities such as sensing and recognizing the environment, position determination, and task execution. The mission planning is considered as the highest level of a hierarchical or layered intelligent control system for an autonomous service robot.

In the following, the state of the art in the motion planning and scheduling of mobile robots as well as for manipulators is presented with particular attention to the advances in mission planning for service robots. Then an integrated approach is presented as a paradigm of optimal multi-target mission planning of a mobile platform in partly known environments with known static obstacles and unknown moving ones, as well as the motion planning of a manipulator mounted on the platform and performing manipulations at the target locations.

ADVANCES AND TRENDS IN MISSION PLANNING OF SERVICE ROBOTS

For systematic and historical reasons the present review starts by presenting motion planning methods for mobile robots and manipulators in known and partly unknown static/dynamic workspaces, followed by scheduling techniques for multi-target routes. This section concludes with methods considering motion planning and scheduling together particularly for service robots. A detailed and systematic presentation of the motion planning problems and the most promising methods that appeared in the relevant literature can be found in two books (LaValle, 2004; Latombe, 1991).

The Motion Planning Problems (MPP) can be distinguished in two main types: (a) static mission planning, that allows a robot to move through stationary obstacles, and (b) dynamic mission planning, which allows a robot to generate a new path or to modify the initial, in response to unknown changes of the environment (Dinham & Fang, 2007).

The appeared approaches for generating paths in static known environments can be classified as either graph-based or artificial potential-field based methods 0. In graph-based methods (Vougioukas, 2005; Pruski & Rohmer, 1997; Song & Amato, 2001; Divelbiss & Wen, 1997; Nearchou, 1998) a graph roadmap (representing the robot's configuration space) is constructed and is searched for deriving the shortest path between start and goal configurations. Potential field methods (Barraquand, et al., 1991; Bemporad, et al., 1996) force the robot to move by the influence of an artificial potential field produced by the goal configuration and the obstacles. The goal configuration generates an "attractive" potential and the obstacles generate a "repulsive" potential. The main limitation arises from the existence of local minima in the resulted field, where no descent direction exists for the robot to follow.

Recently, Xidias and Azariadis (2011) presented a method for solving MPP for a car-like robot moving in 2D terrains. The key-element of this method is the representation of the robot's workspace through a single mathematical entity using the Bump-Surface (B-Surface) concept introduced by Azariadis and Aspragathos (2005). The motion-planning solution is searched on a 2D B-Surface embedded in 3D Euclidean space in such a way that its inverse image into the initial robot's environment satisfies the motion planning objectives and constraints.

In dynamic environments, where particularly service robots are sharing the free space with moving obstacles and/or human beings, the motion planning problem is more challenging. In cases where the trajectory of the moving obstacle is known, the time is defined as an additional workspace dimension. The required path is planned into a formulated space-time of higher dimension where moving obstacles are transformed to static ones (LaValle, 2004). Thus, motion planning among moving and static obstacles is reduced to motion-planning in a stationary environment. Time optimal or near time-optimal approaches (LaValle, 2004) for computing paths through state-time space have been developed, however these algorithms are typically limited to low dimensional state spaces and/or require significant computation time.

In cases where the trajectories of the moving obstacles are unknown, a route is determined offline with one of the aforementioned approaches. Then when changes are observed the robot can be re-planned on the fly (Jaillet & Siméon, 2004; Van den Berg & Overmars, 2005). The Probabilistic Velocity Obstacle approach (Fulgenzi, et al., 2007), is an extension of the local planner to uncertain environments, where uncertainty associated with the obstacle geometry and velocity vector and it is used to artificially grow the velocity obstacle in an open-loop fashion.

Service robots are of immense interest due to their capability to perform complex tasks in many fields such as automated transportation systems in offices, hospitals, libraries, and building management (Schmidt, et al., 1997). The purposes of automation are both to save time and manpower and to improve the service quality. In a market, store, or a library, several stations should be visited by a service robot distributing goods or books to the shelves and an optimal route is required for saving time and expenses. The attainment of this objective necessitates the solution of two known combinatorial optimization problems: (a) motion planning (LaValle, 2004), and (b) vehicle routing and scheduling planning (Qiu, et al., 2002). Both of them are known to be intractable. Motion planning and task scheduling issues are often studied separately. So far, the integration of these problems has been studied by few researchers in Xidias et al. (2009), Herrero-Pérez and Martínez-Barberá (2010), and Xidias and Azariadis (2011) for industrial applications. In Xidias et al. (2009), an Autonomous Guided Vehicle (AGV) is demanded to serve timely (providing delivery tasks) as many work stations in a 2D industrial environment as possible. First, the vehicle's environment is mapped onto a 2D B-Spline surface embedded in 3D Euclidean space using a robust geometric model. Then, a modified genetic algorithm is applied to the generated surface to search for an optimum path that satisfies the requirements of the vehicle's mission. However, this work considers only one moving AGV and does not take into account the corresponding kinematic constraints. In Herrero-Pérez and Martínez-Barberá (2010) a methodology is presented for modeling and controlling a flexible Material Handling System (MHS), composed of AGVs, suitable for flexible manufacturing systems. The AGVs incorporate artificial intelligence and mobile robotics techniques in order to determine their paths. The MHS makes use of a decentralized navigation control and a distributed Petri net in order to achieve higher flexibility and autonomy. However, the method is not globally optimal because the generated paths are not taking into account the task scheduling procedure. In Xidias and Azariadis (2011), a set of AGVs is requested to serve all the workstations cluttered in a 2D environment. Each AGV starts from its depot, passes through a number of workstations (from each one exactly once) and returns back to its depot. The objective is to determine the minimum total travel-time required by the AGVs to serve all workstations in the 2D environment. It must be noticed that, every workstation is allowed to be served by only one AGV. Furthermore, the number and the sequence of the workstations, which are served by a vehicle, are not predetermined. In order to achieve this goal, they utilize the concept of Bump-Surfaces to perform a global search of the solution space in order to ensure an optimal routing-scheduling and motion planning for the set of AGVs moving in the given 2D environment. Finally, the entire problem is formulated as an optimization problem, which is resolved using a GA specifically designed and implemented for the purposes of the current work.

In most of the published works, the Vehicle Routing and Scheduling Problem (VRSP) and Motion Planning Problem (MPP) are studied separately, because the integrated routing-scheduling and motion planning forms a very challenging NP-hard optimization problem (Vis, 2006). VRSP is usually regarded in the literature as a variant of the time-constrained Traveling Salesman Problem (TSP) (Baker, 1983). In VRSP, a single vehicle starts from a depot, visits a set of stations, passing through each one of them exactly once, and returns to the depot; while the overall routing schedule satisfies some predefined time requirements. More general versions of the problem may take into account the capacity constraints of the vehicle or may allow multiple vehicles and time windows (Solomon, 1987). VRSP is usually presented by an undirected graph and its solution is obtained by searching this graph for an optimum route satisfying the related time constraints. All versions of VRSP lead to an NP-hard optimization problem and therefore the trend is to face these problems by using robust heuristics algorithms (Tsitsiklis, 1992) or by using Petri-Net based approaches (Raju & Chetty, 1993; Tatsushi & Ryota, 2010).

A service robot must execute its task with absolute safety and should handle many kinds of objects in a daily life environment. In these cases, the complexity of the motion planning problem is higher since sometimes coordination of the platform and manipulator is necessary. Yamamoto and Yun (1995) studied the problem of navigating a mobile manipulator among obstacles by simultaneously considering the obstacle avoidance problem and the coordination problem. They assume that only the manipulator and not the platform may encounter the obstacle. The proposed controller allows the system to retain optimal or sub-optimal configurations while the manipulator avoids obstacles using potential functions. Tanner and Kyriakopoulos studied the problem of obstacle avoidance by the entire mobile manipulator system (Tanner & Kyriakopoulos, 2000). Their non-holonomic motion planner is based on a discontinuous feedback law under the influence of a potential field.

Recently attempts appeared to study the robot characteristics or to develop robots for transportation and/or distribution of goods in professional environments. Gurcan et al. (2009) investigated the need for automated transportation systems in hospitals. They found that among other alternatives, mobile robots stand out as the most prominent means of automation of transportation tasks in hospitals. An autonomous mobile robotic system with manipulator has been developed to retrieve items from bookshelves and carry them to scanning stations located in the off-site shelving facility (Suthakorn, et al., 2006). In that work the control is considered for navigation as well as for pickup books using the manipulator with high efficiency.

A layered path planning method is presented in partly unknown indoor environment for service robots (Xue & Liu, 2010). A modified Particle Swarm Optimization (PSO) is introduced to determine an initial optimized path in a static workspace. Dynamic layers get multi-pattern information of dynamic obstacles and create a dynamic danger-degree map of the environment. Then a modified A* algorithm is used to avoid dynamic obstacles based on the dynamic danger-degree map. Wosch et al. (2005) introduced a motion planner for service manipulator mounted on a mobile platform interacting with reactive plan execution systems. Collisions are avoided by interacting with an obstacle avoidance system and tactile sensors are used to detect collisions of the manipulator.

The efficiency of the mission planning approaches for service robots depends on the additional criteria and constraints that can be incorporated on top of the obstacle avoidance and optimal path determination such as trajectory smoothness and human safety. Other characteristics of a good motion planning algorithm are the completeness of the solution and the reduce of the complexity and of the required computational time.

In this chapter, we present an approach, which combines some of the positive characteristics of several previous approaches with new ideas to generate an approach that provides an effective solution to the problem of mission planning of a SR moving in an indoor environment. The advantages of the approach are: (a) The SR' path is generated by taking into account the environment's geometry, the depot location, the number and location of predefined stations and the scheduling algorithm. (b) The generated path is smooth and collision-free. (c) The integration of path and velocity planning provides the optimal or near optimal solution for the whole system. In addition, a time optimal algorithm is presented for motion planning of the manipulator for pick and place objects at the stations. The key-element of the approach is the representation of the workspace through a single mathematical entity using the Bump-Surface concept presented in Azariadis and Aspragathos (2005). The entire problem is formulated as a constrained global optimization problem, which is resolved using a Genetic Algorithm (GA) (Goldberg, 1989).

THE MISSION-PLANNING PROBLEM

The presented approach can be applied to a variety of service environments such as stores, supermarkets, hospitals, and libraries where a service robot autonomously transports and manipulates goods. The considered environment, without loose of generality, is a library where the robot transports books from the borrowing desk to the proper bookshelves. It is supposed that the robot's basket is filled with books to be returned to the specified bookshelves by the library personnel and the electronic library management system informs the robot in which bookshelf to put each book. It is assumed that the layout of the library is stored to the robot. The robot should plan an optimal path to transport the books and when it arrives in front of the particular shelf to put the specified book in the bookshelf. In the following, the requirements of the addressed mission planning problem are presented.

Consider a Service Robot (SR) moving in a library environment, in which obstacles (either static or moving) exist. Here, moving obstacles correspond to customers, employees and to any other moving object such as another SR, in the library environment. The set of determined stations $\hat{\mathcal{S}} = \{\mathcal{S}_1,...,\mathcal{S}_m,...,\mathcal{S}_M\}$, $M \geq 1$, represents the desk and bookshelves where the SR should pick up or place books..Figure 1, illustrates a typical example of such dynamic scenario. The overall requirements that must be taken into account are given in the following:

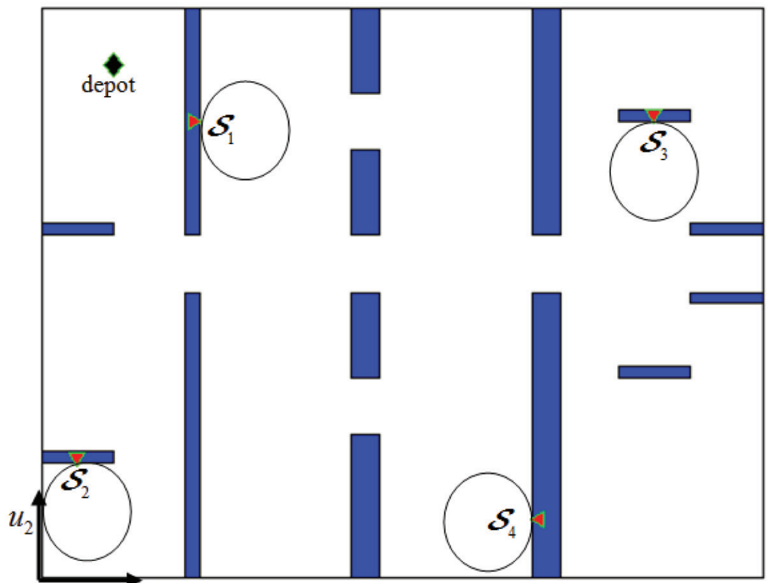
- In order to simplify the representation of the SR' environment, we construct a 2D environment by the projection of the initial 3D environment in the u_1u_2-plane (see Figure 1a).
- The SR is a mobile manipulator.
- The mobile platform is represented by a rectangular-shaped body with two rear wheels and two directional front wheels with a limited steering angle 0. It is equipped with range-sensors encircled around it. The set of sensors defines a region ***RS,*** which is encircled by a circle of radius r_s located in the middle of the robot's body (see the next section ***Definition of a SR***). The set of sensors measures in real-time: the location, the geometry and the instantaneous speed vector (velocity and direction) of the obstacles which are detected by the sensors.
- A PUMA 560 is mounted in the center of the top of the mobile platform.
- The SR is moving only forward with variable velocity in the interval $(0, v_{\max}]$.
- The SR must serve all the given stations and each station should be served only once.
- A SR's path always starts from the library desk (depot), goes through all the stations and terminates at the library desk.
- Each station $\mathcal{S}_m$ is associated with a feasible region, which is represented by a circle, in which the mobile platform can be located to perform a pick and place task without violating the constraints of the manipulator and of the environment (obstacles).
- The moving entities are represented by circular disks.
- The dynamic constraints of the SR are ignored.
- The static obstacles, such as walls and bookshelves, have fixed and known geometry and location.
- The moving obstacles are moving randomly in the environment with unknown trajectories.
- The library desk and the pick and place stations are known a priori.

In the following of this chapter, an approach is presented for the determination of the optimal path for a service robot distributing books and putting them in the right bookshelves taken into account the robot and environment constraints as well as the aforementioned requirements.

CHARACTERISTICS OF A SERVICE ROBOT

The considered service robot in this chapter includes a 6-DoF manipulator mounted on a mobile platform. This section describes the two major components of our SR: the mobile platform and the manipulator.

Figure 1. (a) A projection of a library cluttered with bookshelves, a desk (depot) and 4 stations. It is assumed that the SR should take a book from the stations. The red arrows show the "free" side of the bookshelf. (b) A detailed representation of the initial 3D environment with two bookshelves and books.



(a)

(b)

The Mobile Platform

In order to simplify notation, and without loss of generality, it is assumed henceforth that the 2D environment has unit length in each dimension. Therefore the entire 2D environment is captured by a *normalized workspace* $\mathcal{W} = [0,1] \times [0,1]$. The mobile platform is represented by a car-like robot as it is shown in Figure 2. It has a rectangular body and its motion is bounded by kinematic constraints (LaValle, 2004). The robot's configuration in the 2D environment is uniquely defined by the triple $(u_1, u_2, \theta) \in \mathbb{R}^2 \times [0, 2\pi)$, where $(u_1, u_2) \in \mathcal{W}$ are the coordinates of the rear axle midpoint **R** with respect to a fixed frame, and θ represents the orientation of the vehicle, as it is shown in Figure 2. The steering angle $0 \le \phi \le \phi_{\max}$, where $|\phi| = \arctan\left(\frac{l}{\rho}\right) < \frac{\pi}{2}$, is defined by the main axis of the platform and the velocity vector of the midpoint **F** of the front axis of SR, where ρ is the radius of curvature at point **R** and l is the distance between the midpoints **R** and **F**. **G** is the instantaneous centre of rotation of the platform.

The orientation θ is linked to the derivative at the position of the reference point $R = (u_1, u_2)$ by the equation (LaValle, 2004):

$$\dot{u}_1 \sin\theta - \dot{u}_2 \cos\theta = 0 \qquad (1)$$

Furthermore, a set of range sensors on the perimeter of the platform creates a region ***RS*** which is encircled with a circle of radius r_s and with centre, the centre of the platform. The set of sensors is able to measure in real-time: the location, the geometry and the instantaneous speed vector (velocity and direction) of the obstacles which are detected within range.

It must be noticed that, a path for an autonomous vehicle is a curve in the $u_1u_2\theta$-space that must verify the kinematic constraints (an upper bounded turning radius). However, because of Eq. (1) this path can be also defined by a u_1u_2a-curve defined as $R = R(s)$ (LaValle, 2004).

The Manipulator (PUMA 560)

The manipulator mounted on the center of the mobile platform is a PUMA 560 type manipulator. In the presented approach (and in the simulations), the manipulator is represented by a relatively simpler shape: the first two links of the robot are represented by cylinders with constant radius r_c, the next two links are polyhedral; the 5th link is shown by a cylinder with constant radius r_c, and the hand tip is represented by a polyhedron, as it is illustrated in Figure 3.

OPTIMAL MISSION PLANNING

In this section, an integrated approach is presented for optimal multi-target mission planning of a mobile platform in partly known environments cluttered with known static obstacles and unknown

Figure 2. **A SR in $\mathcal{W}$**

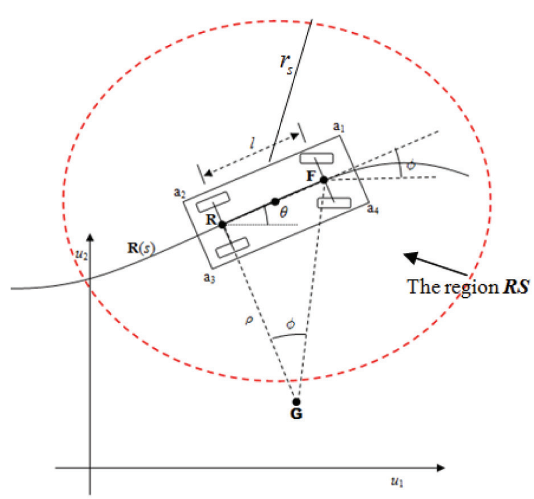
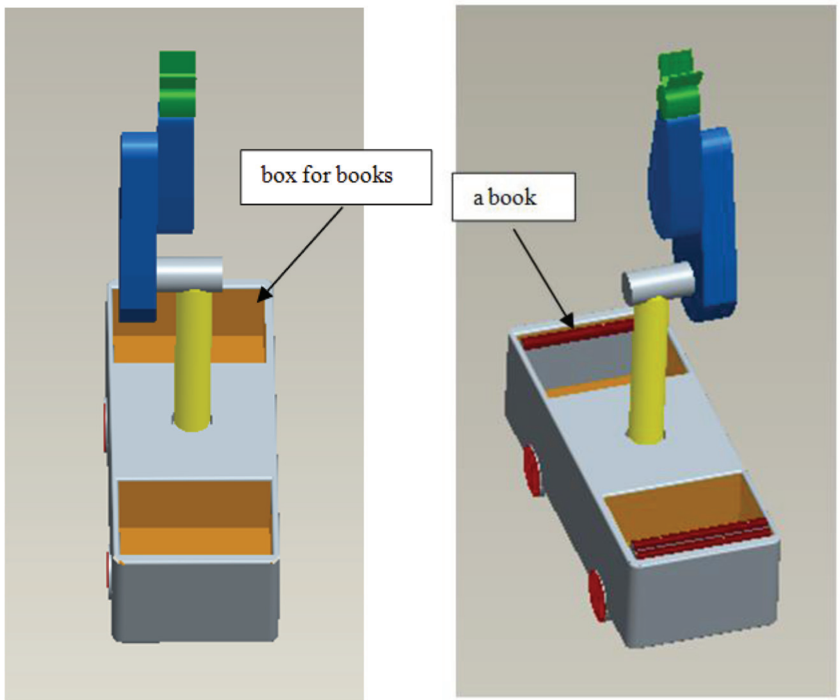


Figure 3. (a) The PUMA 560 mounted on the mobile platform. (b) Another point of view where the SR carries three books.



(a) (b)

moving ones, as well as the motion planning of a manipulator mounted on the platform and performing manipulations at the target locations.

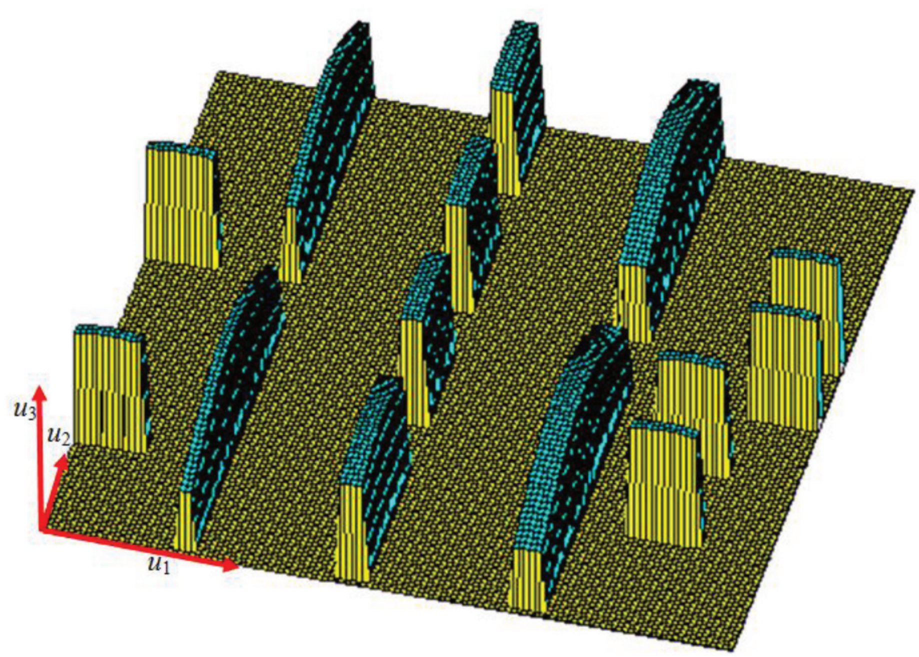
First, the proposed approach computes an optimum path for the SR which connects the depot and the stations by taking into account only the static obstacles of the environment and the dimension of the mobile platform. Then, the SR using a set of onboard range sensors checks if any of the moving obstacles (if exists) is inserted in the region ***RS***. If there are no moving obstacles in the region ***RS*** then the platform moves without modifying its initial path to the next point. If a dynamic obstacle violates the region ***RS*** then a part of its path is modified in order to avoid the collision with the moving obstacle. Finally, when the SR arrives to a station starts to execute its mission. The approach determines a collision free path for the onboard manipulator in order to accomplish its mission. The next sections present the main features of the approach and one case with simulated results to demonstrate the efficiency of this approach.

THE WORKSPACE MODEL

For the representation of the SR' workspace we adopt the method based on the Bump-Surface introduced by Azariadis and Aspragathos (2005). The Bump-Surface concept is a method that represents the entire workspace by using a B-Spline surface embedded in a higher dimension Euclidean Space. For example, the 2D environment shown in Figure 1a is represented by a Bump-Surface embedded in 3D Euclidean space E^3 as it is shown in Figure 4.. The construction of the Bump-Surface is based on a control-points net with variable density depending on the required path-planning accuracy, i.e., denser the grid, higher the accuracy. In addition, due to the flexibility of the B-Spline surfaces we can capture the desired accuracy by taking advantage of their ability for local and global control (Piegl & Tiller, 1997).

Individual motion planning problems can then be formulated as optimization problems where a variety of motion planning objectives and constraints can be encoded in an objective function. The motion planning solution is searched onto the resulted surface (Bump-Surface = Searching space) in such a way that the motion of a SR in the initial environment satisfies the given requirements and constraints. A variety of criteria can be included easily in the objective function taking into account several factors that affect the quality of the solution-path (e.g., kinematical constraints, curvature continuity, robot's dimension, etc.). In contrast to the graph based approaches (i.e., visibility graphs and probabilistic roadmaps) which discretize the vehicle's environment, and the potential field approaches which construct a surface by the union of two mathematical entities (a repulsive potential field and an attractive potential field), the Bump-Surface concept represents the entire vehicle environment by one simple mathematical entity (B-Spline surface) which can be constructed very fast in linear time (Azariadis & Aspragathos, 2005). The resulted surface is free from dead-ends, where a vehicle can be trapped. In fact, the "flat" areas of the Bump-Surface represent the solution space of the problem under consideration while the bumpy areas correspond to prohibited motions. Every feasible path connecting two points onto the "flat" areas of a Bump-Surface avoiding the surface bumps is in

Figure 4. The corresponding bump-surface



fact a solution to the given problem. Since the solution space is limited by the bumpy areas adequate implementations of GAs are utilized in order to find the optimal path corresponding to the problem under consideration. Depending on the motion planning problem a Bump-Surface can have more than three dimensions. For the aforementioned reasons, we adopt in this paper the Bump-Surface method for the representation of the SR's environment. The determination of the Bump-Surface is presented briefly in the following paragraph; however the interested reader can find more details in Xidias and Azariadis (2011), Xidias et al. (2007), and Xidias et al. (2008).

Given a 2D normalized workspace $\mathcal{W}$, the construction of the Bump-Surface is obtained by a straightforward extension of the Z-value algorithm 0. Briefly, this algorithm considers that $\mathcal{W}$ is discretized into uniform subintervals along its u_1 and u_2 orthogonal directions, respectively, forming a grid of points

$$p_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j}) \in [0,1]^3,$$
$$0 \le i, j \le N_g - 1$$

where N_g denotes the grid size. The $z_{i,j}$ coordinate of each grid point $p_{i,j}$ takes a value in the interval $(0,1)$, if the corresponding grid point lies inside an obstacle and the value 0 otherwise.

In this paper, we use a (2, 2)-degree B-Spline surface with uniform parameterization to represent the Bump-Surface $S : [0,1]^2 \to [0,1]^3$, which is given by:

$$S = S(u_1, u_2) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} N_i^2(u_1) N_j^2(u_2) p_{i,j}, \ (u_1, u_2) \in \mathcal{W} \tag{2}$$

where $N_i^2(u_1)$ and $N_j^2(u_2)$ are the B-Spline base functions. The 3D surface S consists of "flat" areas where its third coordinate is zero and "bump" areas which make the length of the vehicles' paths extremely long when they pass through obstacles.

Given a path $R(s)$ and its image $S(R(s))$ on S, the Bump-Surface concept relies on the fact that by construction the arc length of $R(s)$ approximates the arc length of $S(R(s))$ as long as $R(s)$ lies on the flat areas of S (Azariadis & Aspragathos, 2005).

STATIC PATH PLANNING

The main objective of this section is to simultaneously determine the schedule and the path for a SR taking into account only the static obstacles of the environment and the dimension of the mobile platform. The mathematical representation of the SR path should be able to provide simplicity in order to avoid extensive mathematical formulations and derive fast and stable computational algorithms, flexibility, and local control in order to allow for movements within complex environments and narrow passages, accuracy in order to make sure that the SR will pass through the work stations, and so on. For these main reasons, we adopt NURBS to represent the SR's path in this chapter.

A second degree NURBS curve is utilized to represent $R(s)$ by:

$$R(s) = \frac{\sum_{i=0}^{N_c-1} N_i^2(s) w_i p_i}{\sum_{i=0}^{N_c-1} N_i^2(s) w_i}, \ s \in [0,1] \tag{3}$$

Here, $N_i^2(s)$ is the B-Spline basis function, w_i are the weight factors and p_i are the N_c control points of $R(s)$ defined as in the following:

- $p_0 = p_{N_c - 1} = depot$ denoting the depot location of SR.
- $\{p_1, ..., p_{N_c - 2}\} = \begin{pmatrix} \{The\ Work\ Stations\ \mathcal{S}\} \cup \\ \{intermediate\ points\ g_w,\ w = 1, ..., N_b\} \end{pmatrix}$ defined in $\mathcal{W}$.

Note that, $N_b + cardinal(\mathcal{S}) = N_c - 2$. The number N_b is given by $N_b = r(M + 1)$ where r being the number (user-defined) of the points between each pair of $\mathcal{S}_m$. Higher values for r result more flexible paths but the computational time becomes too long. Furthermore, the control points which correspond to the workstations $\mathcal{S}$ are points which located to the centre of the associated circles (Figure 1).

The goal of the proposed mission-planning strategy is the definition of the intermediate points g_w such that the path $R(s)$ satisfies the aforementioned criteria and constraints. For example, in Figure 1, if we set five points between each pair of stations (i.e., $r = 5$) then the total number of the intermediate points g_w becomes $N_b = r(M + 1) = 5(4 + 1) = 25$. An example of a collision free path (a part of the proposed path Figure 7) which starts from depot and passes through the stations Σ_1 and Σ_3 is shown in Figure 5.

Furthermore, it is considered that the orientation of SR in each point of $R(s)$ has the same direction with the tangent vector of $R(s)$ at that point.

Mission Planning for the Mobile Platform

Given the aforementioned formulation, we derive appropriate conditions for the path $R(s)$ in order to satisfy the problem requirements presented in Section *Problem description*.

Conditions for Deriving Collision-Free Paths and for Speed Control

A feasible path $R(s)$ is one that, firstly, does not collide either with the static obstacles and secondly, its curvature $k(s)$ never exceeds an upper-bounded curvature k_{max} in order to satisfy the kinematic constraints and force the platform velocity in the interval $(0, v_{max}]$. Following the results from 0 the arc length of $R(s)$ approximates the length L of its image $S(R(s))$ on S as long as $R(s)$ lies onto the flat areas of S.

In order to take into account the geometry of the mobile platform we select the vertices $\mathbf{a}_k$, $k = 1, ..., 4$, on the perimeter of the platform. Thus, similarly with the midpoint R, each point a_k follows a curve $a_k = a_k(s)$ in $\mathcal{W}$. Then following the results from Xidias et al. (2008), we measure the “flatness” H_k of the image $S_z(a_k(s))$ of $a_k(s)$ on S, i.e., $H_k = \int_0^1 S_z(a_k(s))ds$.

Let $E = e^{\left(\sum_{k=1}^{4} H_k\right)} * L$, be a penalized length function corresponding to $S(R(s))$. E takes a value in the interval $(L, +\infty)$, if the platform collide with the obstacles and the value L, otherwise. Then, the requirement for a collision-free path for the platform can be described as an optimization sub-problem with respect to $p_i = \mathcal{S} \cup \{g_1, ..., g_w, ...g_{N_b}\}$ where the coordinates of stations $\mathcal{S}$ are known but with unknown order of service and the intermediate points g_w, $w = 1, ..., N_b$ have unknown coordinates, written as:

$$\min E \tag{4}$$

Furthermore, in order to ensure that the curvature $k(s)$ along the $R(s)$ never exceeds a maximum curvature $k_{\max}$ to avoid violating the kinematic constraints and force the moving platform velocity in the interval $(0, v_{\max}]$ the following condition should also hold:

$$k(s) \leq k_{\max}\ ,\ s \in [0,1] \tag{5}$$

$R(s)$ is discretized by $N_c - 1$ sequential chords, and therefore the curvature k_i at the R_i point is approximated by the equation (Kobbert, 1996):

$$k_i = \left\| R_{i-1} - 2R_i + R_{i+1} \right\|,\ \ i = 1,\ldots,N_c + 1 \tag{6}$$

Thus, condition (5) is can be rewritten in a discrete manner as:

$$k_i \leq k_{\max}\ ,\ \ i = 1,\ldots,N_c \tag{7}$$

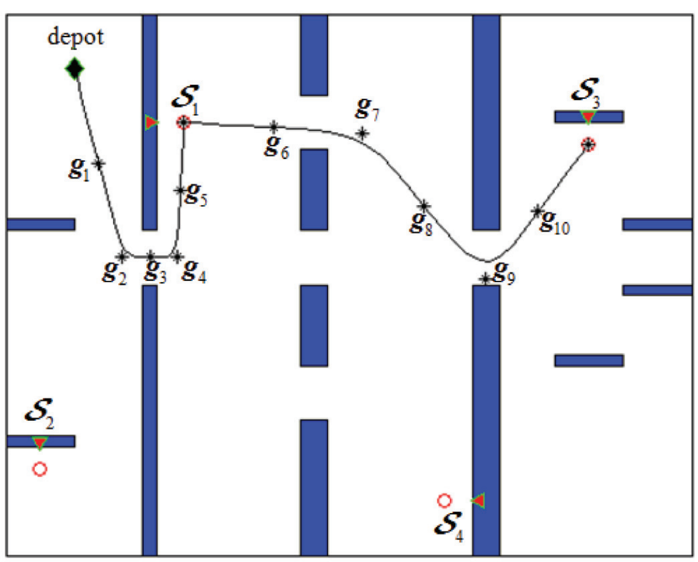
Condition for Deriving the Minimum Travel Time

An optimum velocity profile must be generated for the platform to travel along an assigned path $R(s)$ whose penalized length function onto S is E. Since the main constraint is planning forward motions only, the velocity $v(s)$ is constrained by the relation:

$$0 < v(s) \leq v_{\max} \tag{8}$$

The velocity $v(s)$ can never become negative and can be equal to zero only at the depot locations. Since $R(s)$ is discretized, the measurement of the velocity v_i at every point $\mathbf{R}_i$ is defined by (LaValle, 2004):

Figure 5. A part of the derived path. The black stars show the intermediate points between each pair of stations, including depot.



$$v_i = \begin{cases} v_{\max}, \; if \; k_i = 0 \\ \min\left(v_{\max}, \sqrt{\dfrac{\tau_0}{k_i}} \right), \; if \; k_i \neq 0 \end{cases} \tag{9}$$

where, τ_0 is a constant which depends on the friction between the wheels and the ground, and the gravity constant g.

Let $\mathbf{R}_i$ and R_{i+1} be two sequential points on $R(s)$. Furthermore, it is assumed that the platform is moving from the point $\mathbf{R}_i$ to the point $\mathbf{R}_{i+1}$ in an infinitesimal time Δt_i, and ΔE_i is the corresponding displacement along $R(s)$. The average velocity of the platform during this period is represented by Δv_i. The travel time Δt_i from point R_i to point R_{i+1} is given by:

$$\Delta t_i = \frac{\Delta E_i}{\Delta v_i} \tag{10}$$

Then, the time required for the platform to travel along $R(s)$ is calculated by:

$$t_p = \sum_{i=0}^{N_c - 2} \Delta t_i = \sum_{i=0}^{N_c - 2} \frac{\Delta E_i}{\Delta v_i} \tag{11}$$

The Overall Formulation of the Static Path Planning

Taking the above analysis into consideration, the mission planning problem for the static environment is formulated as an optimization problem given by:

$$\begin{aligned} &\min \left(t_p \right) \\ &subject \; to \; k_i \leq k_{\max}, \; i = 1, \ldots, N_c \end{aligned} \tag{12}$$

The minimization of problem (12) with respect to the control points p_i leads to a collision-free path for the platform, which satisfies all the requirements. The above approach generates for the platform its schedule, the path $R(s)$ and the velocity $v(s)$ simultaneously satisfying that: (a) the platform is moving within the maximum allowed velocity, (b) the platform will not collide with the static obstacles, (c) each work station will be served only once, and (d) the lengths of generated path are the shortest possible.

DETERMINING AN OPTIMUM PATH IN STATIC ENVIRONMENT

Genetic Algorithms (GAs) have been successfully applied to optimization problems with large and complex search spaces due to their ability of reaching a global near-optimal solution even if the search space contains multiple local minima (McCall, 2005). Besides, GAs have extensively been used to solve the motion planning problem (Ali, et al., 2002; Xidias & Azariadis, 2011; Xiao & Michalewicz, 2000), and the routing and scheduling problem (Wang & Tang, 2011). Thus, GAs have been successfully used in the past for the solution of the optimization problems similar to (12).

A modified GA has been designed and implemented to deal with the mission planning problem addressed in this paper. The characteristics of the proposed GA are analyzed and described in following subsections.

The Chromosome Syntax

The first step in applying the GA is the choice of an appropriate representation to encode the decision variables of the problem under consideration. In this work, a mixed integer and floating-point representation was selected for encoding of the variables. That is, chromosomes are strings consisting of a set of successive integers followed by a set of successive real-valued numbers. Each chromosome represents a possible path for the platform in the 2D environment. More specifically, each chromosome consists of $1 + 2N_b + cardinal(\hat{\mathcal{S}})$ genes, where $2N_b$ (2 denotes the dimension of the workspace $\mathcal{W}$) is the number of the intermediate points used for the generated path. The integer left-hand part of the chromosome represents the order with which the platform visits the $\hat{\mathcal{S}}$ work stations Figure 6. demonstrates the structure of the chromosome corresponding to the part of path of Figure 5. As one can be see, the segments of the path between the depot and the stations Σ_1 and Σ_3 are determined by the intermediate points g_w, $w = 1,\ldots,10$.

The Fitness Function

The fitness function evaluates the quality of a chromosome, in other words, the quality of the corresponding path. The fitness function provides the mechanism for determining the direction of the search on the solution space (Bump-Surface). We use the following fitness function:

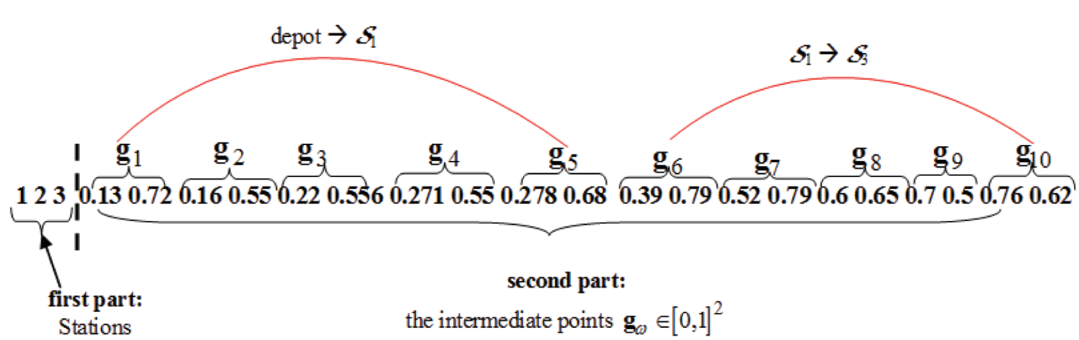
$$\mathcal{F}_p = \begin{cases} \dfrac{1}{t_p}, & if\ k_i \le k_{\max},\ i = 1,\ldots,N_c \\ 0, & otherwise \end{cases} \qquad (13)$$

Each chromosome represents a possible path for the platform as a sequence of control points defining the $R(s)$ curves (see Eq.(3)). The initial population of the proposed GA consists of a number of chromosomes having genes with coordinates randomly selected within $\mathcal{W}$ .Figure 7 shows the solution path which emerges after the implementation of aforementioned approach. The solution path passes through the 4 stations with the order depot-Σ_1-Σ_3-Σ_4-Σ_2- depot.

Genetic Operators

The following three genetic operators were selected for use with the proposed GA: *Reproduction*: In this work, the proportional selection strategy is adopted. According to this strategy, the chromosomes are selected to reproduce their structures in the next generation with a rate proportional to their fitness. *Crossover*: For the first part of the chromosome, that with the integers, the Order Crossover (OX) followed by a suitable repairing mechanism was selected for use, while for the second part of the chromosome, the one-

Figure 6. The chromosome corresponding to the path of Error! Reference source not found.



point crossover was adopted. *Mutation*: For the first part (with integers) the inversion operator is used, while for the second part a boundary mutation was used.

AVOIDING MOVING OBSTACLES

Once the global path $R(s)$ has been created, the robot starts to move along this path in order to serve the stations. If the SR detects a moving obstacle entering in the ***RS*** region another algorithm is activated to modify the initial trajectory. In this part of the chapter the algorithm introduced in Xidias and Aspragathos (2005) for the deviation from the initial path is presented in brief.

At every point $R_i, i = 1, \dots, N_c$ (which correspond to the time instance t_i) of $R(s)$, the SR using the set of the onboard range-sensors checks if any of the moving obstacles are entered in the region ***RS***. If there are no moving obstacles in the region ***RS***, then the robot moves to the next point R_{i+1} of $R(s)$ without modifying its motion. If SR detects a moving obstacle then, taking into account the necessary information of the onboard sensors, it is able to compute the relative velocity $v_{ro}(t_i)$ between the SR and the moving obstacle. By computing the $v_{ro}(t_i)$ we can determine if a collision occurs (for details see Xidias & Aspragathos, 2005). If $v_{ro}(t_i) \le 0$, the SR is moving away from the moving obstacle and no maneuvers are needed. If $v_{ro}(t_i) > 0$, the SR is moving towards to the moving obstacle. In this case, the SR motion should deviate from the initial path in order to avoid collision with the moving obstacle.

Suppose that at time t_i the SR is moving towards the moving obstacle, i.e. $v_{ro}(t_i) > 0$, then, in order for the SR to avoid getting trapped in obstacles' concave regions and bypass any blocking obstacle, the geometry of the moving obstacle it should be modified. The modified obstacle derived from the union of its traces, at the time interval $\left[t_i, t_{i+M}\right]$, where t_{i+M} is the time instance where the SR collides with the obstacle, as illustrated in Figure 8. Then, the Bump-Surface is used in order to determine a "new" path $R'(s)$ for the SR where the initial point is R_i and the final point is R_{N_c}. The local path planning problem is solved using a GA (Azariadis & Aspragathos, 2005). Finally, the SR is moving to the point $R'_1(s)$, which corresponds to the time instance t_{i+1}, and repeats the above procedure. It must be noticed that, in order to ensure that the SR has a smooth motion the following condition is incorporated:

$$\min(\theta(t_{i+1}) - \theta(t_i)) \tag{14}$$

where $\theta(t_i)$ is the SR's orientation at time t_i and $\theta(t_{i+1})$ is the SR's orientation at time t_{i+1}.

MANIPULATOR MOTION PLANNING

When the platform arrives to a station S_m, $m = 1, \dots, M$ stops and starts to execute a predefined task, such as to take a book from the box and put it into the bookshelf. In this section, the formulation of the multi-objective function for the manipulator path planning is described. The problem is treated as a global optimization problem.

Collision Free Motion

This section presents a method for solving the motion-planning problem for the onboard manipulator operating in a 3D environment cluttered with static obstacles (Xidias, et al., 2010). The Bump-Surface is used to formulate a searching space represented by a single mathematical entity, where the optimum sequence of the interme-

Figure 7. The solution path

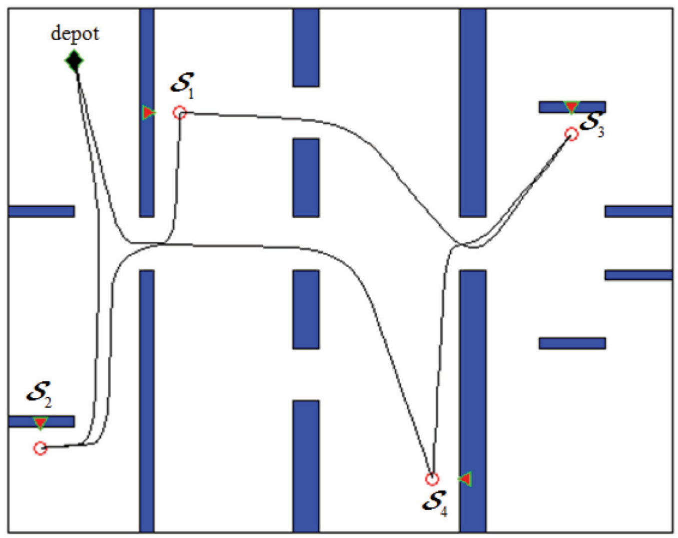
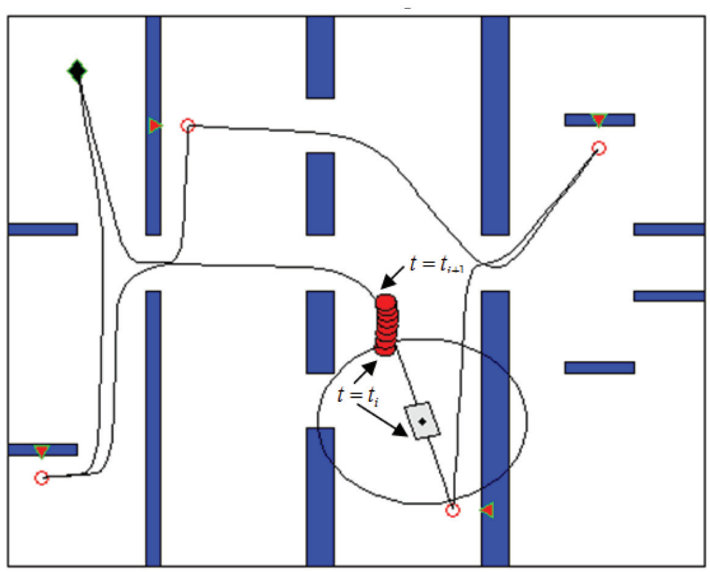


Figure 8. The states of the SR at time instance t_i and the modified obstacle (red color) which derived by the union of its traces at the time interval $\left[t_i, t_{i+M}\right]$. The initial obstacle has circular shape.



diate configurations should be determined towards cycle time minimization and simultaneously a collision-free motion of the manipulator between the obstacles should be obtained. Using the Bump-Surface concept the manipulator's workspace is represented by a 3D surface embedded in $\mathbb{R}^4$, which represents both the free-space and the forbidden areas of the robot's workspace. A global optimization problem is then formulated considering simultaneously the task-scheduling and the collision-free motion planning of the manipulator among the obstacles. The optimization problem is solved using a Genetic Algorithm (GA) with a special encoding that considers the multiplicity of the Inverse Kinematics.

In order to take into account the shape of the manipulator a set of probabilistic points, $a_n^n, n = 1,\dots,N$, defined in the initial 3D environment, is selected on the surface of each *n*-link (*n*=2,…,6) according to the requested accuracy (e.g. higher accuracy is achieved using a big number of N), where N is the overall number of probabilistic points, as illustrated in Figure 9. Thus, following the results from Xidias and Azariadis (2011), each point a_n^n traces a path $a_n^n(s)$ defined in the parametric space $[0,1]^3$. We measure the "flatness" of the image $a_n^n(s)$ on S by:

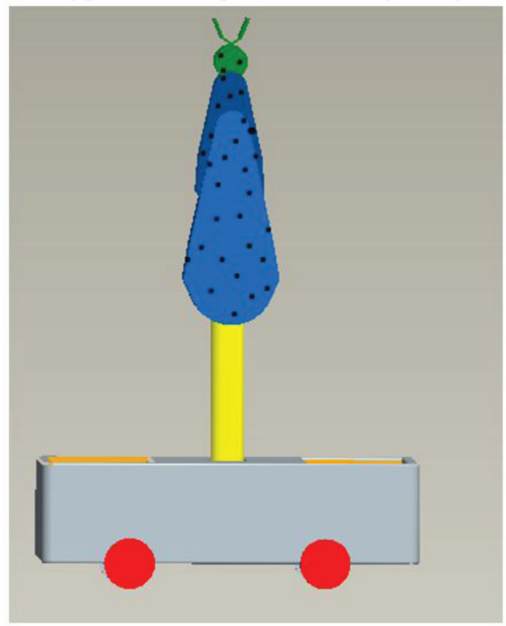
$$H_\mu = \sum_{n=2}^{6} \sum_{n=1}^{N} \left(\int_0^1 S_q \left(a_n^n(s)\right) ds \right) \tag{15}$$

Therefore, the minimization of the following objective function:

$$Flat = \sum_{\mu=1}^{2+\mathcal{R}+\tau} H_\mu \tag{16}$$

with respect to the joint variables $\theta_m \in \mathbb{R}^6, m = 1,\dots,2+\mathcal{R}+\tau$, satisfies the requirement for collision-free robot's configurations, where $2+\mathcal{R}+\tau$ is the total number of configurations between the initial and final configuration and μ is a robot configuration. It is worth noticed that, τ corresponds to the configurations between the successive configurations resulting from linear interpolation of the joint variables and the number $\mathcal{R}$ is the intermediate configurations between two successive task-points specifies the trajectory of the robot while moving between the initial and final configuration. This number is predefined by the operator and it depends from the complexity of the scene; using a large number for $\mathcal{R}$, a tour with higher flexibility is derived, but the computational time is increased.

Figure 9. A set of probabilistic points on the surfaces of the links 2-6



Time Optimal Planning

To fulfill the requirement for time minimization the total travel time of the manipulator should be computed. The travel time for visiting all the stations is determined taking into account the multiplicity of the robot configurations corresponding to each one task-point, as well as the presence of obstacles in the environment. Consequently, the optimum travel time is significantly affected by the configuration choice and the location of the obstacles. It is important to mention that the configuration choices are limited due to the presence of the obstacles.

The time to move the end-effector with a book from the pick point to right point in the bookshelf is divided in the following three parts:

The time t_A spent by the manipulator to travel from the pick configuration to the first intermediate configuration can be written as:

$$t_A = \max_n \left(\frac{\left| q_n^1 - q_n^a \right|}{\dot{q}_n} \right), \qquad n = 1, 2, \ldots, 6 \tag{17}$$

where q^1 expresses the first intermediate configuration, q^a is the configuration corresponding to the pick point and $\dot{q}_n$ is the average velocity of the n_{th}-joint that is assumed to be constant. This approximation is reasonable on condition that the time corresponding to acceleration and deceleration is very small.

It is worth noting that the motion between the pick and place configurations is designated considering linear interpolation of the joint variables. The resulting configurations are also taken into account to assure free motion while moving between the pick and place configurations. The idea behind this approach is the minimization of the number of the intermediate configurations to alleviate the computational burden.

The time $t_{\mathcal{R}}$ spent by the manipulator to travel from one intermediate configuration to another one can be written as:

$$t_{\mathcal{R}} = \sum_{r=2}^{\mathcal{R}} \max_n \left(\frac{\left| q_n^r - q_n^{r-1} \right|}{\dot{q}_n} \right) \tag{18}$$

expressing the time while moving between the intermediate configurations between the initial and final configuration.

The time t_B spent by the manipulator to travel from the last intermediate configuration to the q^b configuration corresponding to the *place point* can be written as:

$$t_B = \max_n \left(\frac{\left| q_n^b - q_n^{\mathcal{R}} \right|}{\dot{q}_n} \right) \tag{19}$$

Thus, the total travel time t_{total} needed to move the manipulator from the pick a book point to the place a book point through the intermediate configurations is given by:

$$t_{total} = t_A + t_{\mathcal{R}} + t_B \tag{20}$$

Thus, the multi-objective function is given by:

$$E_M(q) = w_1\, t_{total}(q) + w_2\, Flat(q) \tag{21}$$

expresses the total cycle time obtained taking into account the initial and final configurations and the $\mathcal{R}$ intermediate configurations and simultaneously ensures that collision avoidance while the manipulator moves between these configurations, where w_1 and w_2 are weight factors with $w_1 + w_2 = 1$ and $w_1, w_2 \geq 0$. The minimization of the multi-objective function E :

$$\min_q \left(E_M \right) = \min_q \left(w_1\, t_{total} + w_2\, Flat \right) \tag{22}$$

with respect to q, where $q = \left(q^{1}, q^{m}, \ldots, q^{2+\mathcal{R}+t}\right)$ provides the solution to the discussed motion planning problem. For the task-points, q expresses a finite number of solutions defined by the inverse kinematics problem, whereas for the intermediate points, q can take an infinite number of values in domain $[0, 2p)$ under the constraint that manipulator have joint limits.

The Optimization Method

The aforementioned motion-planning problem can be characterized as a NP-complete multi-objective optimization problem. Considering that the proposed objective function is procedural, non-continuous, non-linear and multimodal, Genetic Algorithms are selected for the optimization of the multi-goal motion planning.

The representation mechanism: Assume that the PUMA has to visit two configurations in the 3D space with eight (=2^3) configurations corresponding to the initial and final configuration. Each chromosome consists of $3 \times 2 + 6 \times d$ genes, where d is the number of the pick and place configurations of the path between two successive points and is equal to 2. The first part of the string, composed of 2 bytes of 3 bits, represents the manipulator's configuration corresponding to the initial and final point, since each byte of (000, 001, 010, …, 111) determines one out of eight configurations of the manipulator. The second part of the chromosome is composed of $6 \times d$ floating numbers, where each gene stands for a joint angle variable (for details see Xidias, et al., 2010).

The evaluation mechanism: The fitness function, derived from a proper combination of objective functions, is the evaluation mechanism to assess the quality (i.e. the fitness) of each chromosome of the population. The value of the fitness function for one chromosome is the reflection of how well this chromosome is adapted to the environment. This indicates the ability of the chromosome to survive and be reproduced in the next generation. The fitness function of the problem at hand is expressed by:

$$\mathcal{F}_m = \frac{1}{E_M} \tag{23}$$

where the death penalty scheme is applied to handle the constraints and $E_M \succ 0$, since it is a combination of time and distance that cannot be zero.

In the proposed GA, the one-point crossover is used for both the first and second part of the chromosome. *Mutation* is applied in order to inject new genetic material into the population and thereby avoid premature convergence to local minima. For the first part of the chromosome, the mutation operator is applied changing a random gene of digital value '0' to '1' and vice versa. For the second part, the mutation operator is applied changing a random gene (i.e. a floating number) to another one lying in the searching space.

SIMULATED EXPERIMENTS

All simulations are implemented in Matlab and run on a Core 2 Duo 2.13 GHz PC. For the visual representation of the Figures 3, 9, and 12 we used the Pro/Engineer Wildfire 4. In all experiments, the grid size is set to $N_g = 100$, the maximum width is set $w_{\max} = 3$, the minimum width is set $w_{\min} = 0.1$ and the radius of the circular disks representing the moving objects is equal to $r = 0.1$. The number of probabilistic points is set $N = 100$. The settings for the GA's control parameters have been experimentally determined in preliminary tests and defined as follows: population size = 200, maximum number of generations = 500, crossover rate = 0.7, boundary mutation rate = 0.04. Finally, in all experiments a (2, 2)-degree B-Spline surface are used to represent the workspace $\boldsymbol{\Omega}$ and a (2, 2, 2)-degree B-Spline surface are used to represent the onboard manipulator's 3D environment. Due to space limits, we present in this section only one experiment.

Test case: The representative experiment corresponds to a library scenario shown in Figure 1, which is cluttered with narrow corridors, static and one circular moving obstacle. It assumed that the moving obstacle is moving with constant velocity $\left|v_{obs}\right| = 0.5$ and the SR is moving only forward with variable velocity in the interval $(0, 0.5]$. The SR has to travel between a depot and 4 work stations. The number of the unknown control points between the stations is set equal to 5. Thus the overall number of the unknown control points g_w is $N_b = 25$. The solution path is shown in Figure 11. *Eight time instances of the solution motion of the SR*. The magenta dashed curve shows the final path and the black curve shows the initial path, i.e., the path derived by taking into account only the static obstacles and the dimension of the platform. The SR passes through the stations depot-$\boldsymbol{\Sigma}_1$-$\boldsymbol{\Sigma}_3$-$\boldsymbol{\Sigma}_4$-$\boldsymbol{\Sigma}_2$-depot. Figure 11 shows eight time instances of the proposed motion of the SR. The rectangular object (grey color) presents the mobile platform where the black dot represents the onboard manipulator and the red circular disk represent the moving object. The black circle represents the ***RS*** area.

Figure 12 shows the SR in front of the bookshelf (station $\boldsymbol{\Sigma}_1$) where the manipulator gets a book from the box and put it in the bookshelf.

As can see from the above example the proposed method is able to schedule the motion of a SR and simultaneously to produce collision free motion for the onboard manipulator in complicated environments with narrow corridors and rooms. The generated solution path satisfies (in a near optimum way) all the established mission design criteria and constraints. Furthermore, one should bear in mind that we assumed that the mobile platform is car-like robot; hence its motion is

Figure 10. The solution path is represented by the dashed-lines (color magenta), the initial path by a back line.

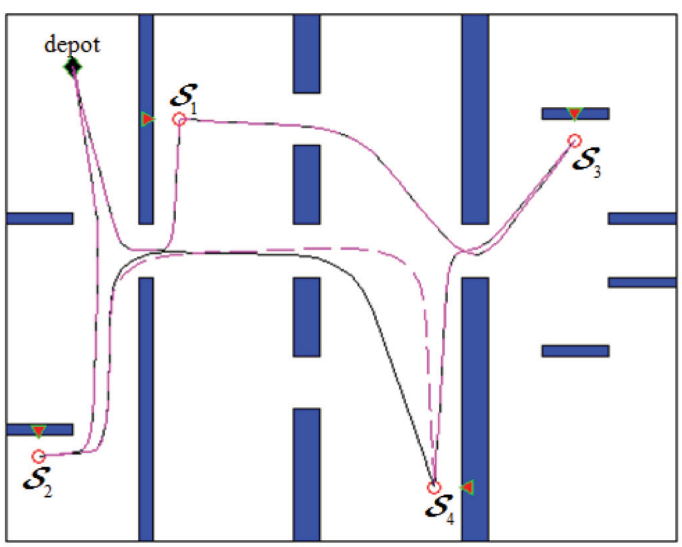


Figure 11. Eight time instances of the solution motion of the SR

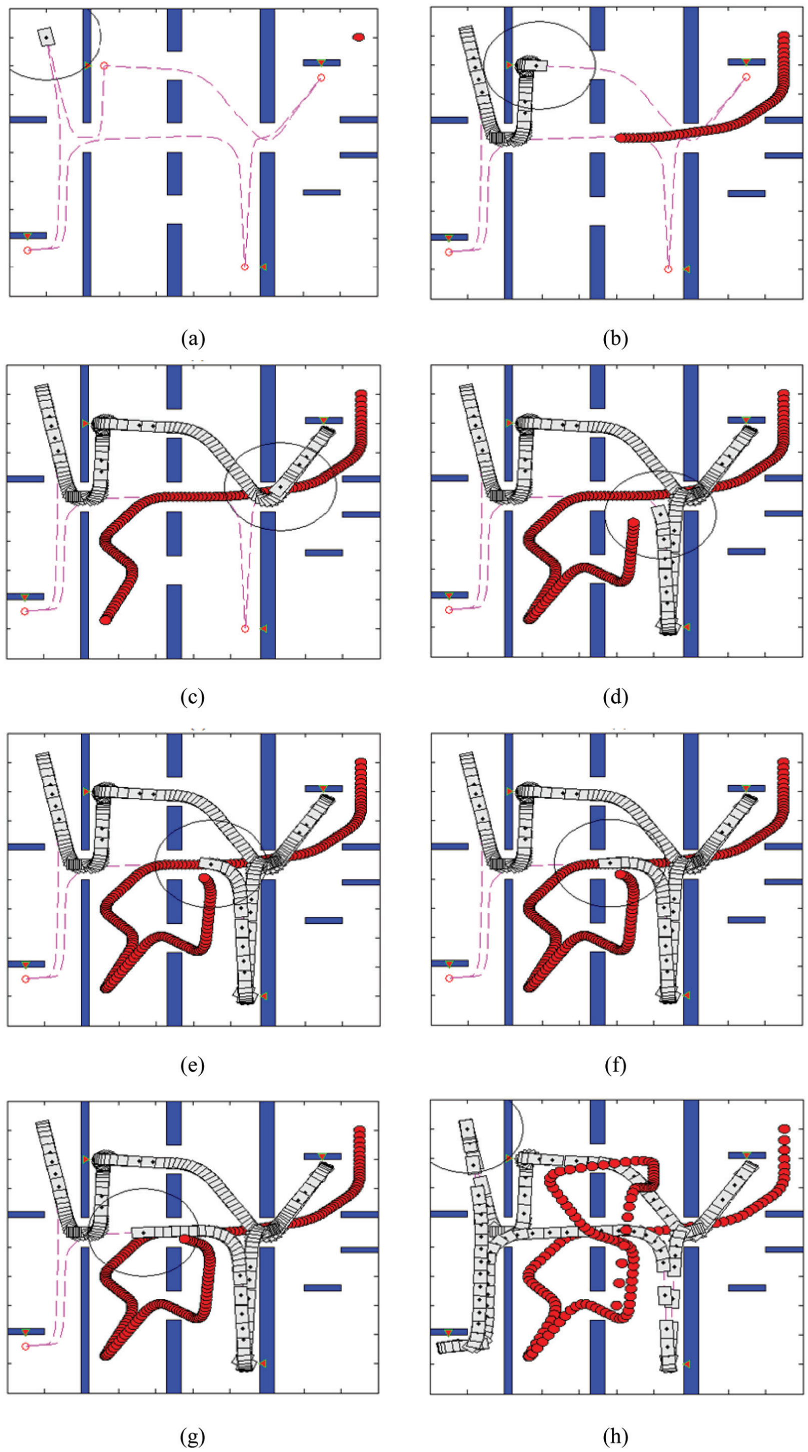
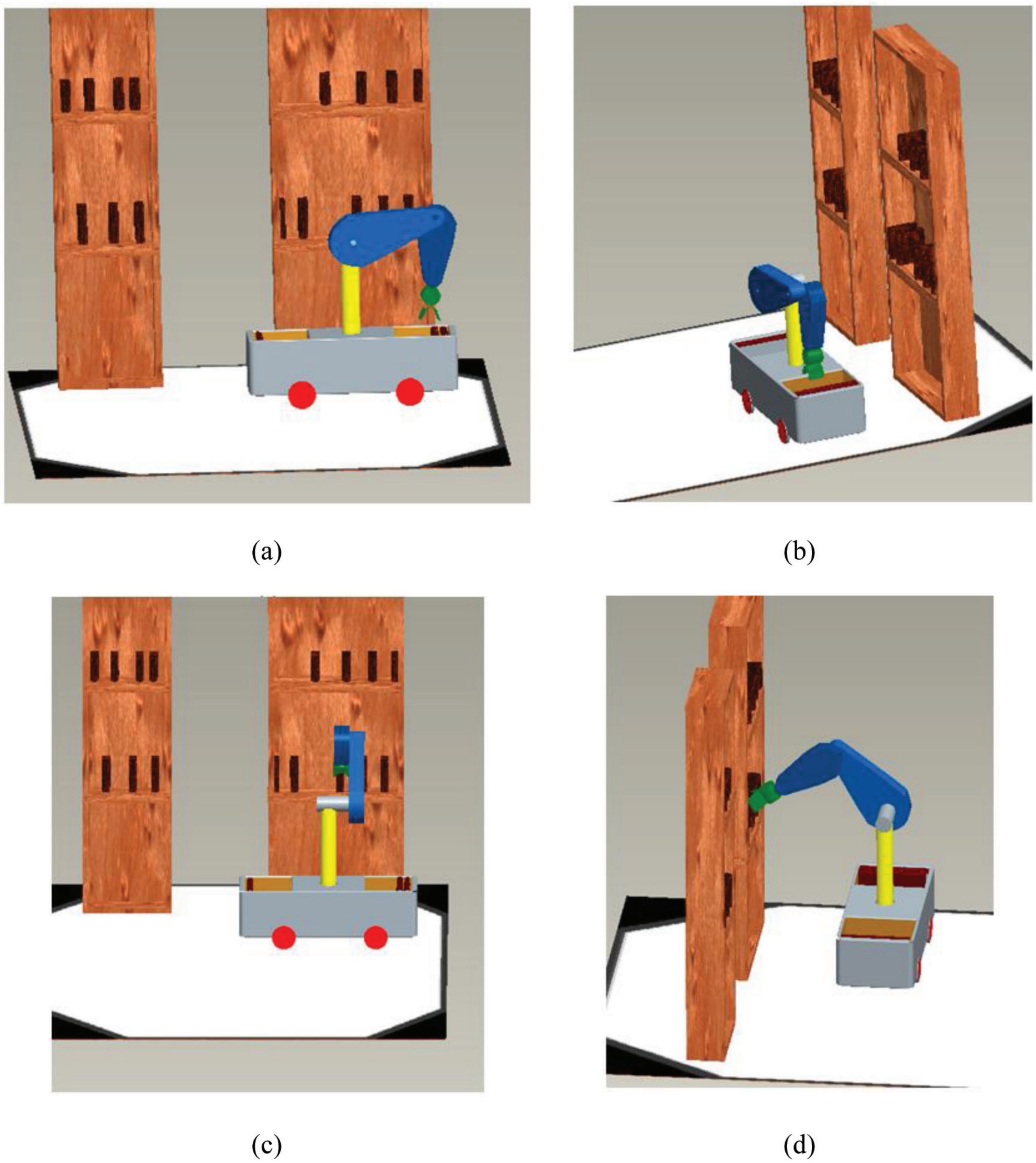
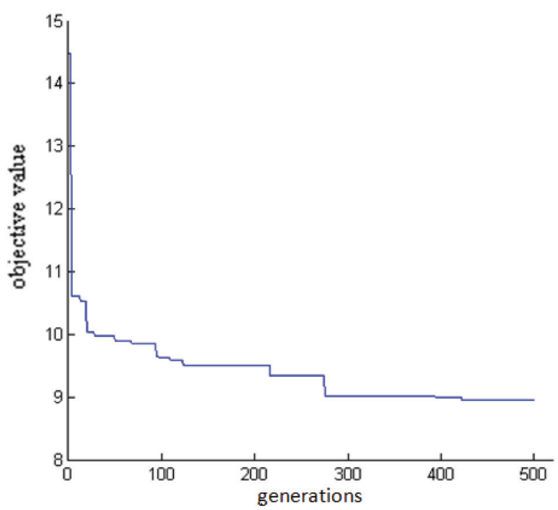


Figure 12. (a) The manipulator taking a book from the box. (b) Another point of view. (c) The manipulator putting a book on the bookshelf. (d) Another point of view.



bounded by kinematic constrains (e.g. an upper bounded steering angle) therefore the motion of the platform is acceptable. In addition, Figure 13 illustrates the convergence of the proposed GA to the global "near" optimum solution in function of the number of generations (see Figure 13).

Figure 13. The convergence of the GA for the problem shown in Figure 10



CONCLUSION

This chapter presents a novel method for mission planning of mobile robots and manipulators for service applications, which address two problems simultaneously: (a) the motion planning problem for a mobile manipulator and (b) vehicle routing and scheduling. The objective is to determine an optimum path for the mobile manipulator so that to serve in minimum travel-time all work stations in a 2D environment, exactly once, while avoiding collisions with the obstacles and each other during their travel. In addition, a time optimal algorithm is presented for motion planning of the manipulator for pick and place objects at the stations. The key-element of the approach is the representation of the workspace through a single mathematical entity using the Bump-Surface concept. The entire problem is formulated as a constrained global optimization problem, which is resolved using a Genetic Algorithm (GA), which utilizes a complex chromosome consisting of integer- and real-value parts. Experiments are conducted showing the effectiveness of the proposed method.

Future work will be concentrated on applying the proposed concept in more complicated environments where a set of mobile manipulators are requested to serve a set of work stations providing pickup and delivery tasks while moving safely (i.e., avoiding any collision with obstacles) in their environment. In addition, transferring the method from the simulation level to the heart of an actual logistics system is a significant issue for a possible future work.

REFERENCES

Ajmal Deen Ali, M. S., Babu, N., & Varghese, K. (2002). Offline path planning of cooperative manipulators using co-evolutionary genetic algorithm. In *Proceedings of the International Symposium on Automation and Robotics in Construction, 19th (ISARC)*, (pp. 415-424). IEEE.

Azariadis, P. N., & Aspragathos, N. A. (2005). Obstacle representation by bump-surfaces for optimal motion-planning. *Robotics and Autonomous Systems*, *51*(2-3), 129–150. doi:10.1016/j.robot.2004.11.001

Baker, E. (1983). An exact algorithm for the time-constrained travelling salesman problem. *Operations Research*, *31*, 938–945. doi:10.1287/opre.31.5.938

Barraquand, J., Langlois, B., & Latombe, J. C. (1991). Numerical potential field techniques for robot path planning. In *Proceedings of the IEEE 5th International Conference on Advanced Robotics*, (pp. 1012-1027). Pisa, Italy: IEEE Press.

Bemporad, A., Luca, A., & Oriolo, G. (1996). Local incremental planning for a car-like robot navigating among obstacles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 1205-1211). Minneapolis, MN: IEEE Press.

Dinham, M., & Fang, G. (2007). Time optimal path planning for mobile robots in dynamic environments. In *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, ICMA 2007*, (pp. 2132-2137). IEEE Press.

Divelbiss, A. W., & Wen, J. T. (1997). A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Transactions on Robotics and Automation*, *13*(3), 443–451. doi:10.1109/70.585905

Fulgenzi, C., Spalanzani, A., & Laugier, C. (2007). Dynamic obstacle avoidance in uncertain environments combining pvos and occupancy grid. In *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE Press.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison Wesley Publishing Company.

Herrero-Pérez, D., & Martínez-Barberá, H. (2010). Modeling distributed transportation systems composed of flexible automated guided vehicles in flexible manufacturing systems. *IEEE Transactions on Industrial Informatics*, *6*(2), 166–180. doi:10.1109/TII.2009.2038691

Jaillet, L., & Siméon, T. (2004). A PRM-based motion planner for dynamically changing environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (vol 2), (pp. 1606-1611). IEEE Press.

Kobbert, L. (1996). A variational approach to subdivision. *Computer Aided Geometric Design*, *13*, 743–761. doi:10.1016/0167-8396(96)00007-6

Latombe, J. C. (1991). *Robot motion planning*. Boston, MA: Kluwer Academic Publishers.

LaValle, M. S. (2004). *Planning algorithms*. Champaign, IL: University of Illinois.

McCall, J. (2005). Genetic algorithms for modeling and optimization. *Journal of Computational and Applied Mathematics*, *184*(1), 205–222. doi:10.1016/j.cam.2004.07.034

Nearchou, A. C. (1998). Path planning of a mobile robot using genetic heuristics. *Robotica*, *16*, 575–588. doi:10.1017/S0263574798000289

Nishi, T., & Maeno, R. (2010). Petri net decomposition approach to optimization of route planning problems for AGV systems. *IEEE Transactions on Automation Science and Engineering*, *7*(3), 523–537. doi:10.1109/TASE.2010.2043096

Özkil, A. G., Fan, Z., Dawids, S., Aanæs, H., Kristensen, K., & Christensen, K. H. (2009). Service robots for hospitals: A case study of transportation tasks in a hospital. In *Proceedings of the IEEE International Conference on Automation and Logistics*. Shenyang, China: IEEE Press.

Piegl, L., & Tiller, W. (1997). *The NURBS book*. Berlin, Germany: Springer-Verlag.

Pruski, A., & Rohmer, S. (1997). Robust path planning for non-holonomic robots. *Journal of Intelligent & Robotic Systems*, *18*, 329–350. doi:10.1023/A:1007937713460

Qiu, L., Hsu, W. J., Huang, S. Y., & Wang, H. (2002). Scheduling and routing algorithms for AGVs: A survey. *International Journal of Production Research*, *40*(3), 745–760. doi:10.1080/00207540110091712

Ravi Raju, K., & Krishnaiah Chetty, O. V. (1993). Addressing design and control issues of AGV-based FMSs with Petri net aided simulation. *Computer Integrated Manufacturing Systems*, *6*(2), 125–134. doi:10.1016/0951-5240(93)90008-E

Rimon, E., & Koditschek, E. (1992). Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, *8*, 501–518. doi:10.1109/70.163777

Schmidt, G., Hanebeck, U. D., & Fischer, C. (1997). A mobile service robot for the hospital and home environment. In *Proceedings of the IARP Second International Workshop on Service and Personal Robots: Technologies and Applications*. Genova, Italy: IARP.

Solomon, M. (1987). Algorithms for the vehicle routing and scheduling problem with time windows constraints. *Operations Research*, *35*, 254–265. doi:10.1287/opre.35.2.254

Song, G., & Amato, N. M. (2001). Randomized motion planning for car-like robots with C-PRM. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (pp. 37-42). IEEE Press.

Suthakorn, J., Lee, S., Zhou, Y., Choudhury, S., & Chirikjian, G. S. (2006). An enhanced robotic library system for an off-site shelving facility. In Yuta, S. (Eds.), *Field and Service Robotics* (pp. 437–446). Berlin, Germany: Springer. doi:10.1007/10991459_42

Tanner, H., & Kyriakopoulos, K. (2000). Nonholonomic motion planning for mobile manipulators. In *Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 1233-1238). IEEE Press.

Tsitsiklis, J. N. (1992). Special cases of travelling salesman and repairman problems with time windows. *Networks*, *22*, 263–282. doi:10.1002/net.3230220305

Van den Berg, J. P., & Overmars, M. H. (2005). Roadmap-based motion planning in dynamic environments. *IEEE Transactions on Robotics*, *21*(5), 885–897. doi:10.1109/TRO.2005.851378

Vis, I. F. A. (2006). Survey of research in the design and control of automated guided vehicle systems. *European Journal of Operational Research*, *170*(3), 677–709. doi:10.1016/j.ejor.2004.09.020

Volpe, R., & Kholsa, P. (1987). Artificial potential with elliptic isopotential contours for obstacle avoidance. In *Proceedings of the 28th IEEE Conference on Decision and Control*, (pp. 180-185). Los Angeles, CA: IEEE Press.

Vougioukas, S. G. (2005). Optimization of robots paths computed by randomized planners. In *Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 2160-2165). Barcelona, Spain: IEEE Press.

Wang, L., & Tang, D. B. (2011). An improved adaptive genetic algorithm based on hormone modulation mechanism for job-shop scheduling problem. *Expert Systems with Applications*, *38*(6), 7243–7250. doi:10.1016/j.eswa.2010.12.027

Wosch, T., Neubauer, W., Wichert, G. V., & Kemeny, Z. (2005). Motion planning for domestic robot assistants. E. Prassler et al. (Eds.), *Advances in Human Robot Interaction*, (pp. 195-205). Berlin, Germany: Springer.

Xiao, J., & Michalewicz, Z. (2000). An evolutionary computation approach to robot planning and navigation. In Hirota, K., & Fukuda, T. (Eds.), *Soft Computing in Mechatronics* (pp. 117–128). Heidelberg, Germany: Springer-Verlag.

Xidias, E., & Aspragathos, N. (2005). Dynamic motion planning for mobile robots using the bump-surface concept. In *Proceeding of the 2nd European Conference on Mobile Robots*. Ancona, Italy: IEEE.

Xidias, E. K., & Azariadis, P. N. (2011). Mission design for a group of autonomous guided vehicles. *Robotics and Autonomous Systems*, *59*(1), 34–43. doi:10.1016/j.robot.2010.10.003

Xidias, E. K., Azariadis, P. N., & Aspragathos, N. A. (2007). Two dimensional motion-planning for nonholonomic robots using the bump-surfaces concept. *Journal of Computing*, *79*(2), 109–118. doi:10.1007/s00607-006-0190-2

Xidias, E. K., Azariadis, P. N., & Aspragathos, N. A. (2008). Path planning of holonomic and non-holonomic robots using bump-surfaces. *Journal of Computer Aided Design and Applications*, *5*(1-4), 497–507.

Xidias, E. K., Nearchou, A. C., & Aspragathos, N. A. (2009). Vehicle scheduling in 2D shop floor environments. *The Industrial Robot*, *36*(2), 176–183. doi:10.1108/01439910910932630

Xidias, E. K., Zacharia, P. T., & Aspragathos, N. A. (2010). Time-optimal task scheduling for articulated manipulators in environments cluttered with obstacles. *Robotica*, *28*(3), 427–440. doi:10.1017/S0263574709005748

Xue, Y.-H., & Liu, H.-P. (2010). Optimal path planning for service robot in indoor environment. In *Proceedings of the International Conference on Intelligent Computation Technology and Automation*. IEEE.

Yamamoto, Y., & Yun, X. (1995). Coordinated obstacle avoidance of a mobile manipulator. In *Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 2255-2260). IEEE Press.